Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждения высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**РАЗРАБОТКА ПЛАГИНА «ЛЮСТРА» ДЛЯ «КОМПАС-3D»**

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

по дисциплине

«Основы разработки САПР» (ОРСАПР)

Выполнил:

студент гр. 580-3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Горюнов А. С.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Руководитель:

к.т.н., доцент каф. КСУП

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Калентьев А.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Томск, 2023

Оглавление

[1 Описание САПР 3](#_Toc148454878)

[1.1 Описание программы 3](#_Toc148454879)

[1.2 Описание API 4](#_Toc148454880)

[1.3 Обзор аналогов плагина 6](#_Toc148454881)

[2 Описание предмета проектирования 8](#_Toc148454882)

[3 Проект системы 10](#_Toc148454883)

[3.1 Диаграмма классов 10](#_Toc148454884)

[3.2 Макеты пользовательского интерфейса 11](#_Toc148454885)

[4 Список источников 12](#_Toc148454886)

1. **ОПИСАНИЕ САПР**
   1. **Описание программы**

КОМПАС-3D – это российская импортонезависимая система трехмерного проектирования, ставшая стандартом для тысяч предприятий и сотен тысяч профессиональных пользователей.

КОМПАС-3D широко используется для проектирования изделий основного и вспомогательного производств в таких отраслях промышленности, как машиностроение (транспортное, сельскохозяйственное, энергетическое, нефтегазовое, химическое и т.д.), приборостроение, авиастроение, судостроение, станкостроение, вагоностроение, металлургия, промышленное и гражданское строительство, товары народного потребления и т. д. [1]

**«Компас 3D» включает в себя множество полезных функций для работы над инженерными проектами:**

* **параметрическая и твердотельная разработка**, позволяющая при помощи эскизов создавать модель, к которой применяются все основные свойства софта;
* **библиотека стандартных моделей** — позволяет использовать для разработки встроенный каталог простых деталей;
* **2D проектирование** — создание чертежей и технической документации проекта;
* **использование листового материала** — проектирует детальные изделия, включая изгибы, резьбу, вырезу, отверстия;
* **учет допусков** — учитывает усадку, свойства и параметры материалов, а также технологию производства окончательного проекта;
* **инструментарий** — включает обширный набор инструментов, включая изменение размеров, геометрию объекта, шероховатость. [2]

Аналогом являетсяAutodesk Inventor. Это – это программа для проектирования всех типов изделий промышленного производства на основе их параметров. Характеристики объектов определяют математические модели, любое изменение которых автоматически влияет на конфигурацию. [3]

Основными преимуществами Inventor являются:

* Богатые возможности интеграции с другими продуктами Autodesk, такими как AutoCAD, Vault, и многими другими;
* Autodesk Inventor имеет более широкое мировое сообщество пользователей и доступ к множеству онлайн-ресурсов и форумов для получения поддержки и помощи.
  1. **Описание API**

API КОМПАС3D – это ориентированные на прикладного программиста инструментальные средства разработки приложений (библиотек конструктивов, прикладных САПР) на базе системы КОМПАС. API КОМПАС3D включает в свой состав 2D API и 3D API. [4]

Используемые классы: ksPart – это класс в библиотеке Компас 3D API, который представляет собой 3D-деталь в приложении Компас 3D; ksRadialArray – этот класс позволяет создавать радиальные массивы объектов вокруг выбранной оси.

Таблица 2.1 – Используемые свойства класса Application

|  |  |  |
| --- | --- | --- |
| Название | Тип данных | Описание |
| Name | string | Название детали |
| Model | ksModel | 3D-модель детали. Возвращает объект ksModel, представляющий 3D-модель детали. |

Таблица 2.1 – Продолжение

|  |  |  |
| --- | --- | --- |
| VariableTable | ksVariableTable | Таблица переменных и параметров детали. Возвращает объект ksVariableTable, который позволяет управлять переменными и параметрами детали. |

Таблица 2.2 – Используемые метода класса Application

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| Open | string fileName |  | Открывает существующую деталь по указанному пути fileName. |
| Save |  |  | Сохраняет деталь |
| GetDefaultEntityParams | int type | ksEntityParam | Возвращает объект, представляющий параметры для создания геометрических объектов (например, линий, окружностей) определенного типа (указанным type). |

Таблица 2.2 – Используемые метода класса Application

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Входные параметры | Тип возвращаемых данных | Описание |
| SetParam | ksEntity axis, int type, int count, double step |  | Устанавливает параметры радиального массива, такие как объект axis, тип массива, количество копий (count) и угловой шаг (step). |
| AddElements | int count |  | Добавляет элементы к массиву, увеличивая количество копий. |
| SetType | int type |  | Устанавливает тип массива. |

* 1. **Обзор аналогов плагина**

Косвенным аналогом для плагина “Люстра” является плагин GearTeq для SolidWorks.

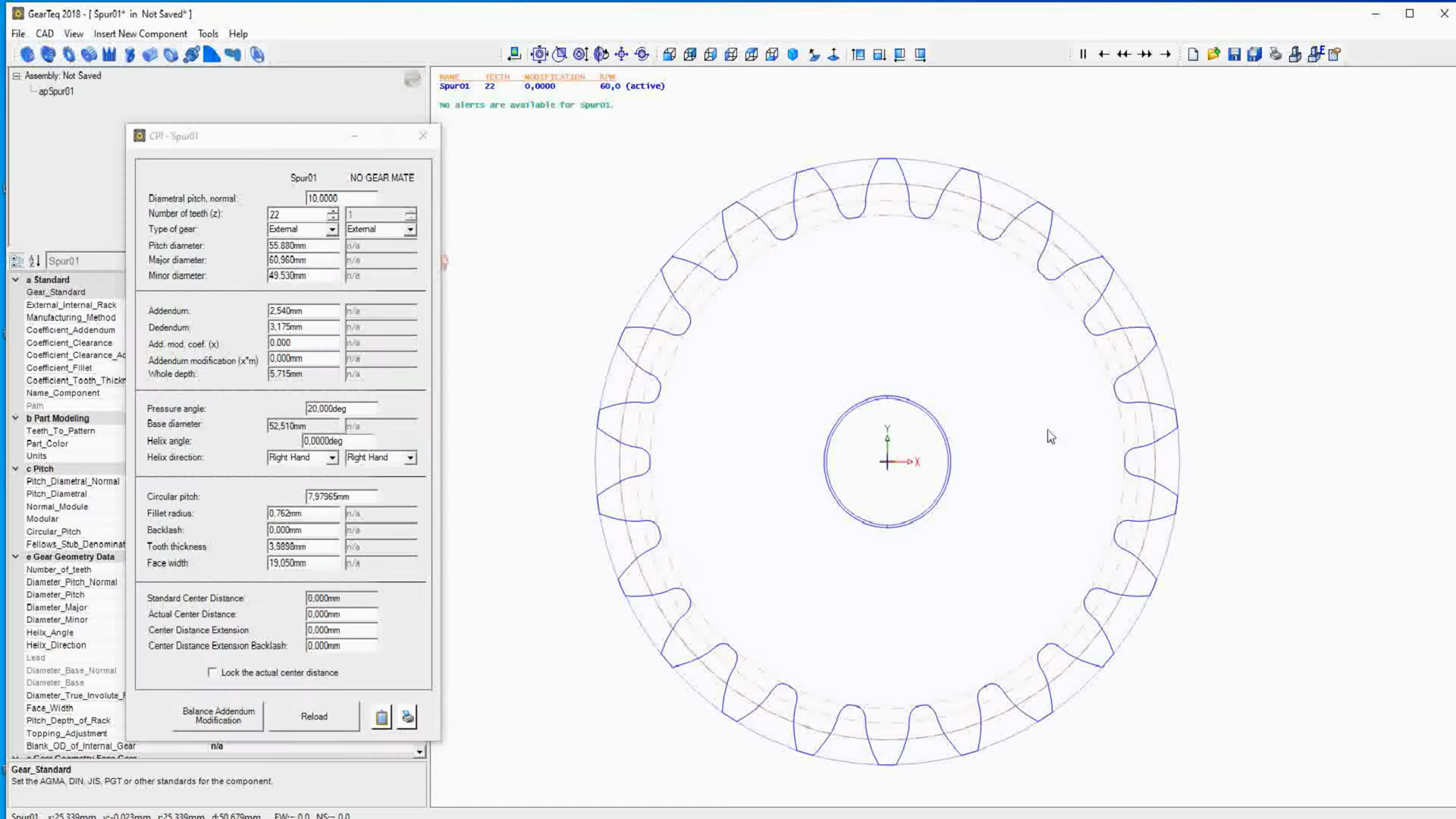


Рисунок 1 – Пользовательский интерфейс GearTeq

Этот плагин можно считать аналогом так как в нем есть множество настраиваемых свойств, и можно добиться создания основы под люстры. Затем подредактировать ее, и получить что-то похожее на люстру.

Однако, этот плагин все-таки косвенный, так как он предназначен совершенно под другие задачи, а именно для проектирования различных шестеренок. [5]

1. **ОПИСАНИЕ ПРЕДМЕТА ПРОЕКТИРОВАНИЯ**

Люстра — подвесной потолочный светильник для создания бестеневого, в отличие от точечного источника, освещения помещений. Размеры люстры (расстояние между отдельными точечными источниками) и количество точечных источников света в ней определяют равномерность освещения. [6]

Назначение разрабатываемого плагина обусловлено быстрым моделированием люстр разных типов. Благодаря данному расширению, мастера по люстрам могут наглядно рассмотреть спроектированную модель, при необходимости перестроить под необходимые им параметры. На рисунке 2 представлена модель люстры.

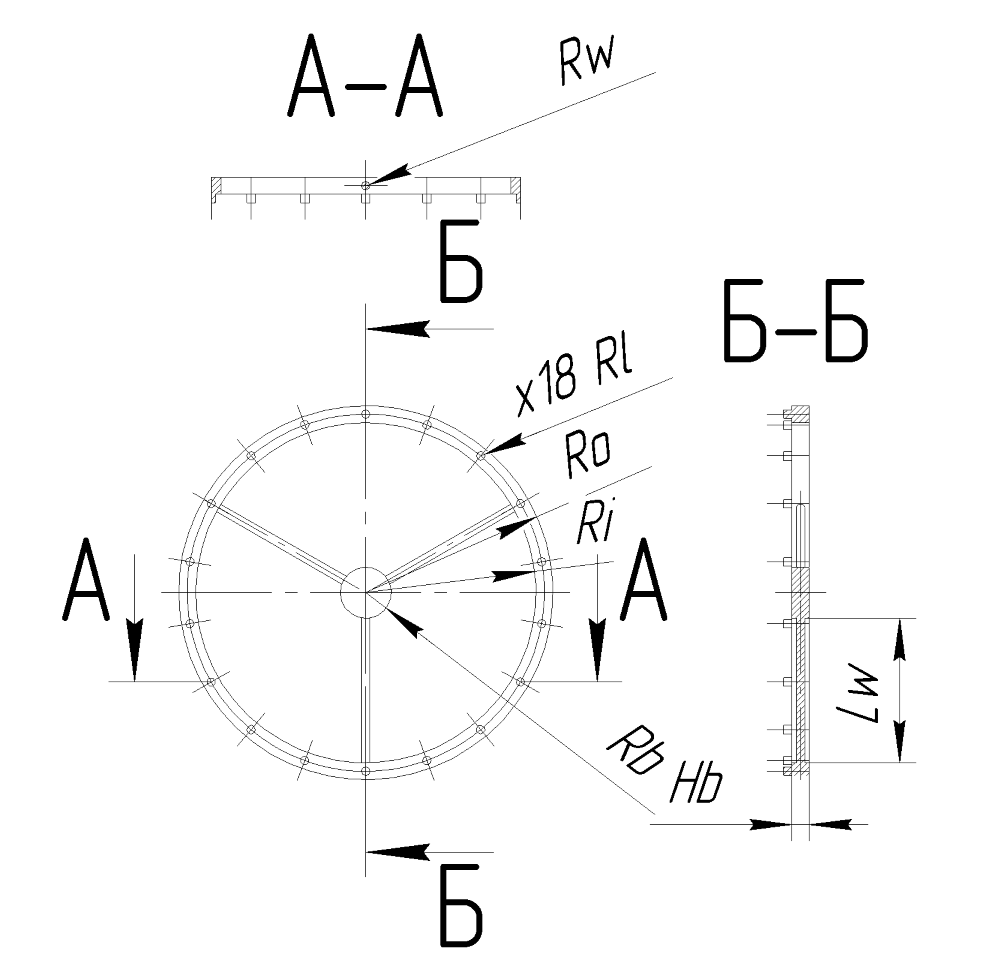


Рисунок 2 — Модель люстры с размерами

Система должна быть выполнена в одном из двух вариантов:

* В качестве встроенного плагина САПР “ КОМПАС-3D”, который запускается непосредственно из САПР;
* В качестве сторонней программы, способной запустить процесс программы “КОМПАС-3D” для построения детали.

Изменяемые параметры для плагина (также все обозначения показаны на рис. 2.1):

* Радиус внешней окружности люстры Ro (от 400 до 1000 мм);
* Радиус внутренней окружности Ri (от Rb + 50 до Ro – 50 мм);
* Радиус основания люстры окружности Rb (от 100 до 200 мм);
* Толщина окружностей Hb (от 40 до 80 мм);
* Длина труб под провода Lw, измеряется автоматически;
* Радиус окружности трубы под провода Rw, измеряется автоматически;
* Количество лампочек n (2nRl < 2πRi);
* Радиус лампочек Rl (от 15 до 25 мм).

АС должна иметь пользовательский интерфейс с возможностью изменения значений, представленных выше, и последующим построении объекта «Люстра» в САПР КОМПАС-3D. В плагине должны проходить проверки значений, вводимых пользователем. Реализуемый плагин должен обеспечивать обработку ошибочных ситуаций, возникающих в процессе работы. При нажатии на кнопку «Построить» должна проходить проверка правильности ввода данных. Если данные некорректные, то должно высветиться окно с ошибкой построения и не будут применяться введенные параметры.

1. **ПРОЕКТ СИСТЕМЫ**
   1. **Диаграмма классов**

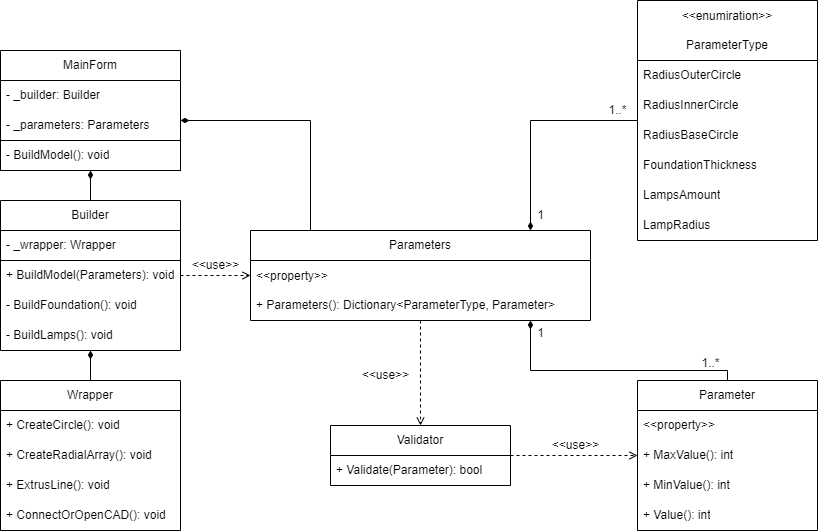


Рисунок 3 — Диаграмма классов

Разберем основные классы проекта:

− MainForm – является главным окном приложения. Хранит в себе параметры (Parameters) и объект класса строителя модели (Builder);

− Parameters – класс, хранящий в себе все параметры модели;

− Buider – класс строитель модели;

− Wrapper – класс обертка API САПР. В нем находятся все нужные методы создания примитивов и документов, которые пригодятся для построения модели;

− ParameterType – перечисление параметров детали;

− Parameter – Значения свойств параметра;

− Validator – Класс для проверки значений.

* 1. **Макеты пользовательского интерфейса**

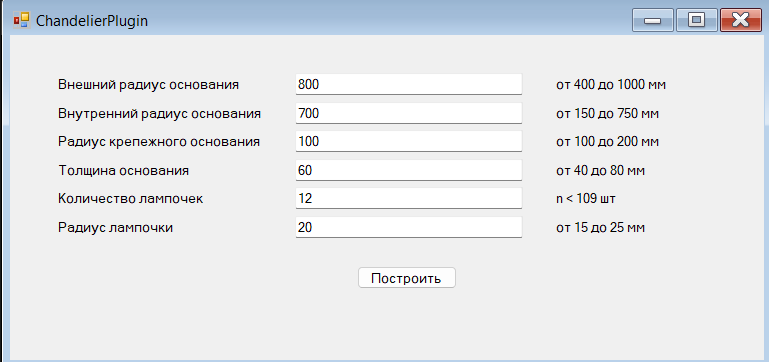
****

Рисунок 4 — Пользовательский интерфейс

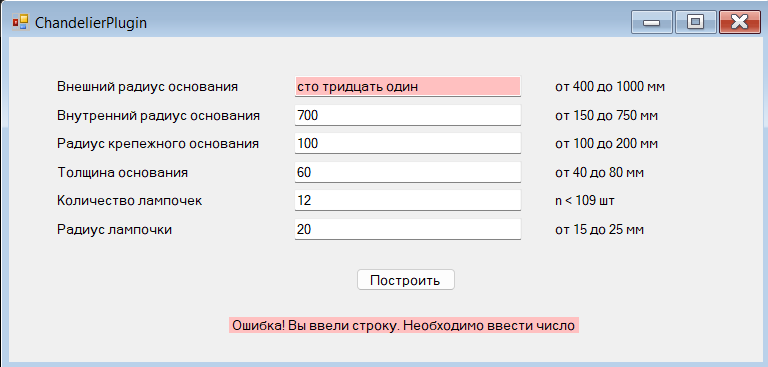
****

Рисунок 5 — Пользовательский интерфейс с неправильно введенным параметром

**Список источников**

* + - 1. О программе KOMPAS-3D [Электронный ресурс] // KOMPAS-3D [Официальный сайт] URL: <https://kompas.ru/kompas-3d/about/> (дата обращения: 17.10.2023);
      2. "KOMPAS-3D: основные возможности и преимущества" [Электронный ресурс]: [Статья] // junior3d.ru - уроки и материалы для 3ds max. URL: <https://junior3d.ru/article/Kompas-3D.html> (дата обращения: 17.10.2023);
      3. "Основные возможности Autodesk Inventor" [Электронный ресурс]: [Статья] // Информационное агентство avtosreda.ru. URL: <https://avtosreda.ru/info/osnovnye-vozmozhnosti-autodesk-inventor/> (дата обращения: 17.10.2023);
      4. "Руководство по KOMPAS-Invisible" [Электронный ресурс]: PDF-документ. URL: <https://kompas.ru/source/documents/2021/%D0%A0%D1%83%D0%BA%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE%20KOMPAS-Invisible.pdf> (дата обращения: 17.10.2023);
      5. Gearteq для SolidWorks [Электронный ресурс] // Страница партнера SolidWorks [Официальный сайт] URL: <https://www.solidworks.com/ru/partner-product/gearteq> (дата обращения: 17.10.2023);
      6. Люстра [Электронный ресурс] // Страница на Википедии URL: <https://ru.wikipedia.org/wiki/Люстра> (дата обращения: 17.10.2023).